

Docket No. AUS920010231US1

**CLAIMS:**

What is claimed is:

1. A method for detecting a termination of a process within a plurality of processes in a data processing system, the method comprising the steps of:
  - establishing, within the plurality of processes, a monitoring policy, wherein the monitoring policy assigns a first process within the plurality of processes to monitor a second process within the plurality of processes;
  - responsive to a termination of execution of the second process, determining the cause of the execution termination by the first process; and
  - responsive to a determination that the second process terminated execution in an abnormal manner, attempting to restart the second process by the first process.
2. The method as recited in claim 1, further comprising:
  - opening a communications link between the first process and the second process; and
  - posting a blocking read by the first process in order to detect termination of the second process.
3. The method as recited in claim 2, wherein the communications link is a watch FIFO (first-in first-out) communications link.

Docket No. AUS920010231US1

4. The method as recited in claim 3, wherein posting a blocking read to the watch FIFO communications link is performed in a single thread of execution.

5. The method as recited in claim 2, wherein the first process posts a blocking operation on the communications link that completes the blocking operation on termination of the second process.

6. The method as recited in claim 2, wherein the blocking operation is performed in a separate execution thread within the first process in order to allow normal process operation of the second process to continue.

7. The method as recited in claim 1, further comprising:

responsive to a determination that the second process terminated execution in a normal manner, modifying the monitoring policy, wherein the monitoring policy is modified to assign the first process to monitor a third process within the plurality of processes; and monitoring of the third process by the first process for termination of execution.

8. The method as recited in claim 1, wherein the plurality of processes forms a ring of plurality of processes.

9. A method for inserting a process within a plurality of processes containing a first process and a monitoring policy in a data processing system, the method comprising the steps of:

Docket No. AUS920010231US1

receiving a request for a second process to join the plurality of processes;

responsive to the second process joining the plurality of processes, selecting the first process  
5 within the plurality of processes to monitor the second process; and

modifying the monitoring policy, wherein the selected first process monitors the second process for termination of execution.

10 10. The method as recited in claim 9, wherein the second process monitors the process previously monitored by the selected first process.

11. The method as recited in claim 9, wherein the second process joining the plurality of processes joins as a  
15 child of the first process.

12. The method as recited in claim 9, wherein the request is received by a inter-process communications (IPC) mechanism.

13 The method as recited in claim 9, wherein the  
20 plurality of processes forms a ring of plurality of processes.

14. The method as recited in claim 9, further comprising:

locking a state file in a join communications link;  
25 and

blocking any other processes attempting to simultaneously join the plurality of processes.

093643-05404

Docket No. AUS920010231US1

15. The method as recited in claim 14, wherein the communications link is a first-in first-out communications link.

16. The method as recited in claim 14, wherein blocking  
5 any other processes attempting to simultaneously join the plurality of processes is by a file lock.

17. A data processing system for detecting a termination of a process within a plurality of processes in a data processing system, comprising:

10 establishing means for establishing, within the plurality of processes, a monitoring policy, wherein the monitoring policy assigns a first process within the plurality of processes to monitor a second process within the plurality of processes;

15 determining means, responsive to a termination of execution of the second process, for determining the cause of the execution termination by the first process; and

20 attempting means, responsive to a determination that the second process terminated execution in an abnormal manner, for attempting to restart the second process by the first process.

18. The data processing system as recited in claim 17, further comprising:

25 opening means for opening a communications link between the first process and the second process; and  
posting means for posting a blocking read by the first process in order to detect termination of the second process.

098643 0340  
04250 374960

Docket No. AUS920010231US1

19. The data processing system as recited in claim 18, wherein the communications link is a watch FIFO (first-in first-out) communications link.

20. The data processing system as recited in claim 19,  
5 wherein posting a blocking read to the watch FIFO communications link is performed in a single thread of execution.

21. The data processing system as recited in claim 18,  
10 wherein the first process posts a blocking operation on the communications link that completes the blocking operation on termination of the second process.

22. The data processing system as recited in claim 18,  
15 wherein the blocking operation is performed in a separate execution thread within the first process in order to allow normal process operation of the second process to continue.

23. The data processing system as recited in claim 17, further comprising:

20 modifying means, responsive to a determination that the second process terminated execution in a normal manner, for modifying the monitoring policy, wherein the monitoring policy is modified to assign the first process to monitor a third process within the plurality of processes; and

25 monitoring means for monitoring of the third process by the first process for termination of execution.

Docket No. AUS920010231US1

24. The data processing system as recited in claim 17, wherein the plurality of processes forms a ring of plurality of processes.

25. A data processing system for inserting a process  
5 within a plurality of processes containing a first process and a monitoring policy in a data processing system, the method comprising the steps of:

receiving means for receiving a request for a second process to join the plurality of processes;

10 selecting means, responsive to the second process joining the plurality of processes, for selecting the first process within the plurality of processes to monitor the second process; and

modifying means for modifying the monitoring policy,  
15 wherein the selected first process monitors the second process for termination of execution.

26. The data processing system as recited in claim 25, wherein the second process monitors the process previously monitored by the selected first process.

20 27. The data processing system as recited in claim 25, wherein the second process joining the plurality of processes joins as a child of the first process.

28. The data processing system as recited in claim 25, wherein the request is received by a inter-process  
25 communications (IPC) mechanism.

0964490540

Docket No. AUS920010231US1

29. The data processing system as recited in claim 25, wherein the plurality of processes forms a ring of plurality of processes.

30. The data processing system as recited in claim 25,  
5 further comprising:

locking means for locking a file used to join the communications link; and

blocking means for blocking any other processes attempting to simultaneously join the plurality of  
10 processes.

31 . The data processing system as recited in claim 30, wherein the communications link is a FIFO communications link.

32. The data processing system as recited in claim 30,  
15 wherein blocking any other processes attempting to simultaneously the plurality of processes is by a file lock.

33. A data processing system, comprising:  
a bus system;  
20 a memory, containing a set of instructions and a first and second processes within a plurality of processes, connected to the bus system; and  
a processing unit connected to the bus system,  
wherein the processing unit executes the set of  
25 instructions from the memory to establish, within the plurality of processes, a monitoring policy, wherein the monitoring policy assigns the first process within the plurality of processes to monitor the second process

Docket No. AUS920010231US1

within the plurality of processes, responsive to a termination of execution of the second process, the processing unit instructs the first process to determine the cause of the execution termination, and responsive to  
5 a determination that the second process terminated execution in an abnormal manner, the processing unit instructs the first process to attempt to restart the second process.

34. A data processing system, comprising:  
10 a bus system;  
a memory, containing a set of instructions and a first process within a plurality of processes and a second process, connected to the bus system; and  
a processing unit connected to the bus system,  
15 wherein the processing unit executes the set of instructions from the memory to receive a request for the second process to join the plurality of processes, responsive to the second process joining the plurality of processes, the processing unit selects the first process  
20 within the plurality of processes to monitor the second process, and the processing unit instructs the first process to modify the monitoring policy, wherein the selected first process monitors the second process for termination of execution.

25  
35. A computer program product in a computer-readable medium for detecting a termination of a process within a plurality of processes in a data processing system, comprising:  
30 instructions for establishing, within the plurality of processes, a monitoring policy, wherein the monitoring



Docket No. AUS920010231US1

policy assigns a first process within the plurality of processes to monitor a second process within the plurality of processes;

instructions, responsive to a termination of  
5 execution of the second process, for determining the cause of the execution termination by the first process; and

instructions, responsive to a determination that the second process terminated execution in an abnormal  
10 manner, for attempting to restart the second process by the first process.

36. The computer program product as recited in claim 35, further comprising:

instructions for opening a communications link  
15 between the first process and the second process; and  
instructions for posting a blocking read by the first process in order to detect termination of the second process.

37. The computer program product as recited in claim 36,  
20 wherein the communications link is a watch FIFO communications link.

38. The computer program product as recited in claim 37,  
wherein the instructions for posting a blocking read to the watch FIFO communications link is performed in a  
25 single thread of execution.

39. The computer program product as recited in claim 36, wherein the first process posts a blocking operation on

2025 RELEASE UNDER E.O. 14176

Docket No. AUS920010231US1

the communications link that completes the blocking operation on termination of the second process.

40. The computer program product as recited in claim 36, wherein the blocking operation is performed in a separate execution thread within the first process in order to allow normal process operation of the second process to continue.

41. The computer program product as recited in claim 35, further comprising:

10 instructions, responsive to a determination that the second process terminated execution in a normal manner, for modifying the monitoring policy, wherein the monitoring policy is modified to assign the first process to monitor a third process within the plurality of processes; and

15 instructions for monitoring of the third process by the first process for termination of execution.

42. The computer program product as recited in claim 35, wherein the plurality of processes forms a ring of plurality of processes.

43. A computer program product for inserting a process within a plurality of processes containing a first process and a monitoring policy in a data processing system, comprising:

25 instructions for receiving a request for a second process to join the plurality of processes;

instructions, responsive to the second process joining the plurality of processes, for selecting the

0986448 0540  
T04250 8749850

Docket No. AUS920010231US1

first process within the plurality of processes to monitor the second process; and

instructions for modifying the monitoring policy, wherein the selected first process monitors the second  
5 process for termination of execution.

44. The computer program product as recited in claim 43, wherein the second process monitors the process previously monitored by the selected first process.

45. The computer program product as recited in claim 43,  
10 wherein the second process joining the plurality of processes joins as a child of the first process.

46. The computer program product as recited in claim 43, wherein the request is received by a inter-process communications (IPC) mechanism.

15 47. The computer program product as recited in claim 43, wherein the plurality of processes forms a ring of plurality of processes.

48. The computer program product as recited in claim 43, further comprising:

20 instructions for locking a state file in a join first-in first-out communications link (FIFO); and  
instructions for blocking any other processes attempting to simultaneously join the plurality of processes.

2025 RELEASE UNDER E.O. 14176

Docket No. AUS920010231US1

49. The computer program product as recited in claim 48, wherein the communications link is a FIFO communications link.

50. The method as recited in claim 48, wherein blocking  
5 any other processes attempting to simultaneously join the plurality of processes is by a file lock.

FILED OCT 14 1992